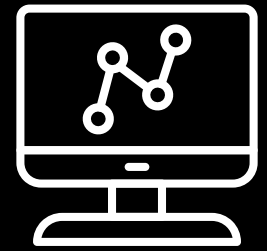


PROJET DÉVELOPPEMENT LOGICIEL

PERROT Loick



L'objectif de ce projet était de développer une application de gestion pour une chaîne de restaurants fictive. La demande consistait à équiper chaque établissement d'une tablette dotée de notre logiciel. Ses fonctionnalités incluaient la gestion des stocks ainsi que la surveillance en temps réel de la consommation énergétique des appareils et de la température des locaux via différents capteurs.

Rapport de faisabilité/cahier des charges

Dans cette optique, nous avons débuté par l'élaboration d'un rapport de faisabilité et d'un cahier des charges. Cette étape nous a permis d'analyser la viabilité de chaque exigence de l'entreprise et de définir les solutions techniques pour les réaliser.



Rapport de faisabilité :

Résumé des critères demandés :

La chaîne de restaurants « Cook & Save » souhaite maîtriser ses **stocks alimentaires** et l'optimisation de sa **consommation énergétique**. La direction souhaite mettre en place une application centralisée pour suivre précisément les stocks et les ventes afin d'améliorer l'efficacité de ses restaurants, et de monitorer la consommation électrique des appareils principaux (fours, réfrigérateurs, climatisation) pour identifier les sources d'inefficacité.

Objectifs du projet :

- Suivi des achats : enregistrement des livraisons (légumes, viande, uranium, produits laitiers...)

- Suivi des ventes : Saisie des menus et des produits les plus populaires dans les restaurants.
- Suivi du stock disponible : avec une interface simple à comprendre et à utiliser ainsi que la possibilité d'ajouter de nouveaux produits à la liste de ceux déjà existants.
- L'application doit être disponible sur tablette.
- Collecte des données pour le suivi énergétique : les données sont récoltées à l'aide de capteurs individuels sur chaque appareil de la cuisine.
- Alerte immédiate en cas de panne d'un appareil, surtout pour les frigos.
- Détection des appareils dont la consommation est trop élevée par rapport à la moyenne des autres restaurants.
- Tableaux de bord permettant de comparer facilement les performances de chaque restaurant.
- Aide à la décision en fournissant des données et des analyses objectives pour justifier le remplacement de certains équipements.

Analyse de la faisabilité :

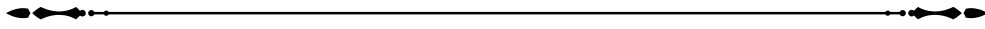
Pour la gestion des stocks, créer une application mobile connectée à une base de données centrale qui recense les stocks disponibles et permet de le modifier. La complexité de la création d'une telle application est relativement faible, il suffit de faire une interface simple à utiliser. Les employés devront faire attention à ne pas se tromper en saisissant les stocks. Il faut s'assurer d'avoir un serveur en état disponible pour stocker les données. **Faisabilité de l'application de gestions des stocks : OK à condition que les serveurs soient fiables**

Pour le suivi des ventes, il serait intéressant d'intégrer un logiciel directement dans les caisses afin qu'elles enregistrent dans la base de données les menus et les ingrédients associés. Ainsi, des statistiques pourront être faites sur ces données et des tableaux de bords seront créés. **Faisabilité de l'interface permettant les statistiques sur les ventes de produits : OK**

Pour le suivi énergétique, les capteurs présents dans les machines vont récolter les informations sur la consommation et sur l'état de fonctionnement de l'appareil. La difficulté va être de définir des seuils à partir desquels on peut affirmer qu'une machine est en panne. Les capteurs de température des bâtiments seront aussi utilisés. **Faisabilité de la récolte des données énergétiques et de la détection des pannes : OK**

Conclusion :

Le projet commandité rentre totalement dans nos capacités. Les principales contraintes sont liées à la simplicité de l'interface utilisateur et à la fiabilité du matériel à disposition (serveurs, capteurs, tablettes...). **Le bilan final est positif.**



Cahier des charges :

1.1 Contexte du projet :

L'entreprise Cook & Save qui gère une chaîne de restaurants souhaite moderniser la manière dont elle gère ses établissements afin d'optimiser ses dépenses et garantir le bon fonctionnement de ses appareils électriques. Actuellement, l'entreprise gère ses stocks et sa consommation d'énergie de manière dispersée, avec un mélange d'équipements connectés et non connectés. Nous allons donc proposer une application accessible sur tablette permettant de :

- Suivre les stocks et les ventes pour améliorer l'efficacité
- Suivre la consommation d'énergie et l'état des appareils pour localiser les pannes

1.2 Besoins fonctionnels :

Fonctionnalités MVP :

Ces fonctionnalités sont indispensables pour que le système fonctionne.

Pour les employés :

- Consulter l'état des stocks disponibles en temps réel.
- Enregistrer les entrées de stocks.
- Saisir les sorties de stock liées aux ventes.
- Ajouter de nouveaux produits à la liste existante.

Pour les gérants de site :

- Recevoir une alerte immédiate en cas de panne d'un appareil.

Pour le système :

- Récolter automatiquement les données de température et de consommation grâce aux capteurs.

- Détecter les appareils dont la consommation dépasse la moyenne.

Fonctionnalités secondaires. Ces fonctionnalités peuvent apporter des capacités d'analyse et de prise de décision.

Pour les gérants de sites :

- Visualiser des tableaux des graphiques qui mettent en avant les performances énergétiques des différents restaurants.

- Consulter les analyses pour justifier le remplacement des équipements.

1.3 Contraintes :

Contraintes matérielles :

- L'application doit fonctionner sur des tablettes (4Go de RAM et 128Go de stockage minimum).

- Les capteurs et thermomètres doivent être connectés et fonctionnels (en remplaçant les équipements non connectés).

- Le serveur doit disposer d'une solution de backup.

Contraintes techniques :

- Les sites doivent disposer d'un réseau wifi privé stable.

- Le serveur doit être capable de stocker la totalité des données recueillies et de les restituer rapidement.

Organisationnelles :

- Les employés et les gérants doivent être formés à l'utilisation de l'application.

- L'entreprise doit gérer elle-même l'organisation avec les réparateurs pour les interventions.

1.4 Critères de succès :

Stabilité technique : Le serveur de backup doit être disponible en cas de panne du serveur principal.

Performances : Tous les capteurs doivent être connectés et en bon état.

Respect du budget : Le coût de déploiement de tous ces éléments ne doit pas dépasser les 2000€ de budget annoncé.

2. UML : Spécifications

Cette partie détaille la modélisation fonctionnelle du système Cook & Save.

2.1 Cas d'utilisation

Voici la liste des actions principales possibles pour chacun des acteurs liés à notre projet.

Pour les employés :

- Consulter les stocks et visualiser la liste et la quantité des stocks disponibles.
- Enregistrer une livraison en saisissant les entrées de marchandises.
- Ajouter de nouveaux produits si nécessaire.

Pour les gérants des sites :

- Consulter le tableau de bord énergétique pour comparer les performances.
- Recevoir des alertes en cas de panne.
- Identifier les appareils qui consomment trop d'énergie.
- Commander du nouveau matériel.

Pour le système :

- Collecter les données des capteurs connectés.
- Détecter les anomalies en comparant les données aux seuils fixés.

2.2 Scénarios textuels

Détail du déroulement des cas d'utilisation critiques pour assurer la bonne compréhension du fonctionnement de l'application.

Cas d'utilisation : Enregistrer une livraison de stock

Ce scénario est très important pour la fiabilité de l'inventaire.

- **Titre du cas** : Enregistrer une livraison.

- **Acteur principal** : Employé.

- **Préconditions** : L'employé est authentifié sur la tablette et la connexion au serveur est active.

- **Scénario nominal** :

1. L'employé sélectionne l'option "Réception de commande".
2. Le système affiche la liste des fournisseurs ou des catégories de produits.
3. L'employé sélectionne le produit livré
4. L'employé saisit la quantité reçue.
5. Le système demande une confirmation.
6. L'employé valide.
7. Le système met à jour le stock dans la base de données centrale.

- **Scénarios alternatifs** :

- **Produit inconnu** : L'employé ne trouve pas le produit. Il sélectionne "Ajouter nouveau produit", saisit les informations, et le système l'ajoute à la liste existante.

- **Postconditions** : Le stock est actualisé.

Cas d'utilisation : Gestion automatique d'une panne (Suivi énergétique)

Ce scénario couvre l'aspect critique de la sécurité alimentaire.

- **Titre du cas** : Détection et alerte de panne frigo.

- **Acteur principal** : Système (Capteurs/Serveur).

- **Préconditions** : Les capteurs sont connectés et fonctionnels.

- **Scénario nominal** :

1. Le capteur du frigo relève une température supérieure au seuil défini.
2. Le capteur transmet la donnée au serveur.
3. Le serveur analyse la donnée et confirme l'anomalie (panne potentielle).
4. Le serveur génère une notification d'alerte.
5. Le système envoie l'alerte sur la tablette ou l'ordinateur du Gérant de site.

- **Scénarios alternatifs** :

o Perte de connexion capteur : Le serveur ne reçoit pas de données pendant un laps de temps défini. Il génère une alerte "Capteur déconnecté".

• **Postconditions** : L'incident est ouvert et le gérant est informé pour action.

2.3 Diagrammes de séquence

Ce diagramme illustre les interactions techniques pour la consultation des stocks, vérifiant la fiabilité du serveur et du réseau.

Exemple textuel (Flux d'interaction) : Consulter le Stock

1. **Employé** -> **Tablette (Interface)** : Appuie sur le bouton "Voir Stocks".
2. **Tablette** -> **Serveur Central** : Envoie une requête dans la base *Stocks*.
3. **Serveur Central** -> **Base de Données** : Interroge la table *Stocks*.
4. **Base de Données** -> **Serveur Central** : Retourne la liste des produits et quantités.
5. **Serveur Central** -> **Tablette** : Renvoie les données.
6. **Tablette** -> **Employé** : Affiche la liste des produits sous forme de tableau lisible.

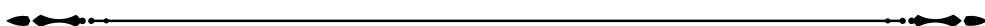
3. Synthèse

Ce cahier des charges met en avant la réponse aux besoins de la chaîne « Cook & Save » en structurant le projet autour de deux piliers : la maîtrise des stocks et l'efficacité énergétique.

Le projet est techniquement réalisable et cohérent avec les capacités de l'équipe. Il repose sur :

- Une architecture client-serveur robuste (Tablettes -> Serveur Central -> Capteurs).
- Un périmètre matériel défini (Serveurs avec backup, tablettes performantes, capteurs connectés) respectant les contraintes budgétaires estimées entre 840€ et 2000€.
- Une gestion des risques anticipée, notamment via la mise en place de backups et l'utilisation de matériel fiable pour pallier aux pannes.

Le respect de ces spécifications garantira une transition fluide, permettant à Cook & Save d'atteindre ses objectifs d'optimisation.



Maquette / POC

1 – Maquette fonctionnelle

Notre application sera divisée en trois onglets distincts. Deux pour les gérants de sites et un pour les employés chargés de gérer les stocks.

L'interface est très simple. Une fois l'application ouverte, les trois onglets sont accessibles en un clic.

• Parcours type pour un gérant de site :

Le gérant de site commence par ouvrir l'application. En haut de la page, trois onglets se présentent. S'il souhaite visualiser la consommation des appareils, il lui suffit d'appuyer sur cet onglet. Une fois sur cette page, il peut sélectionner à l'aide de listes déroulantes l'appareil dont il veut visualiser les données.

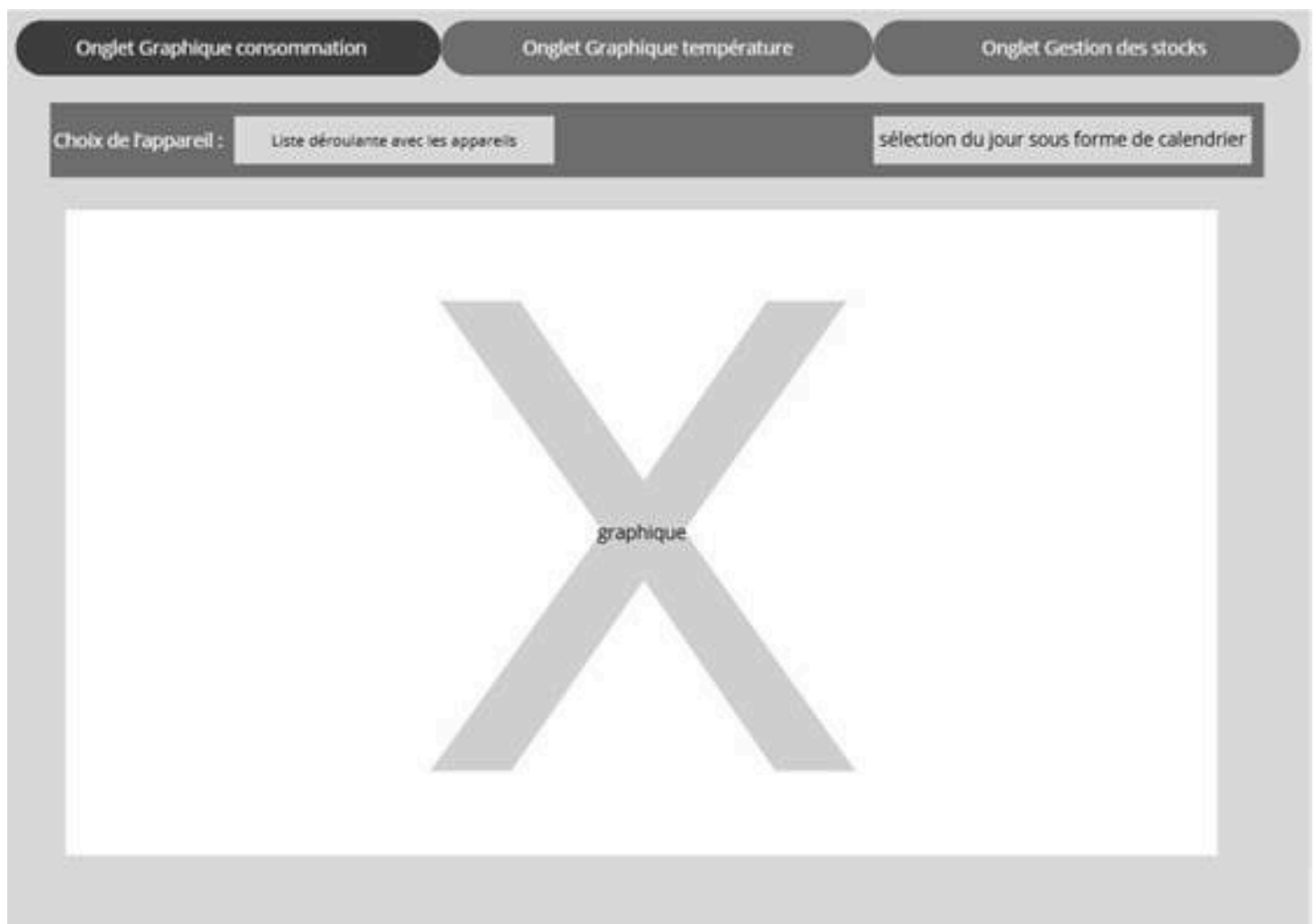


Figure 1 : Onglet consommation

Le principe est le même pour la visualisation des données de température. Le gérant de site peut cliquer sur l'onglet de la température (celui du milieu) et sélectionner un capteur ou une

sonde pour visualiser les données.

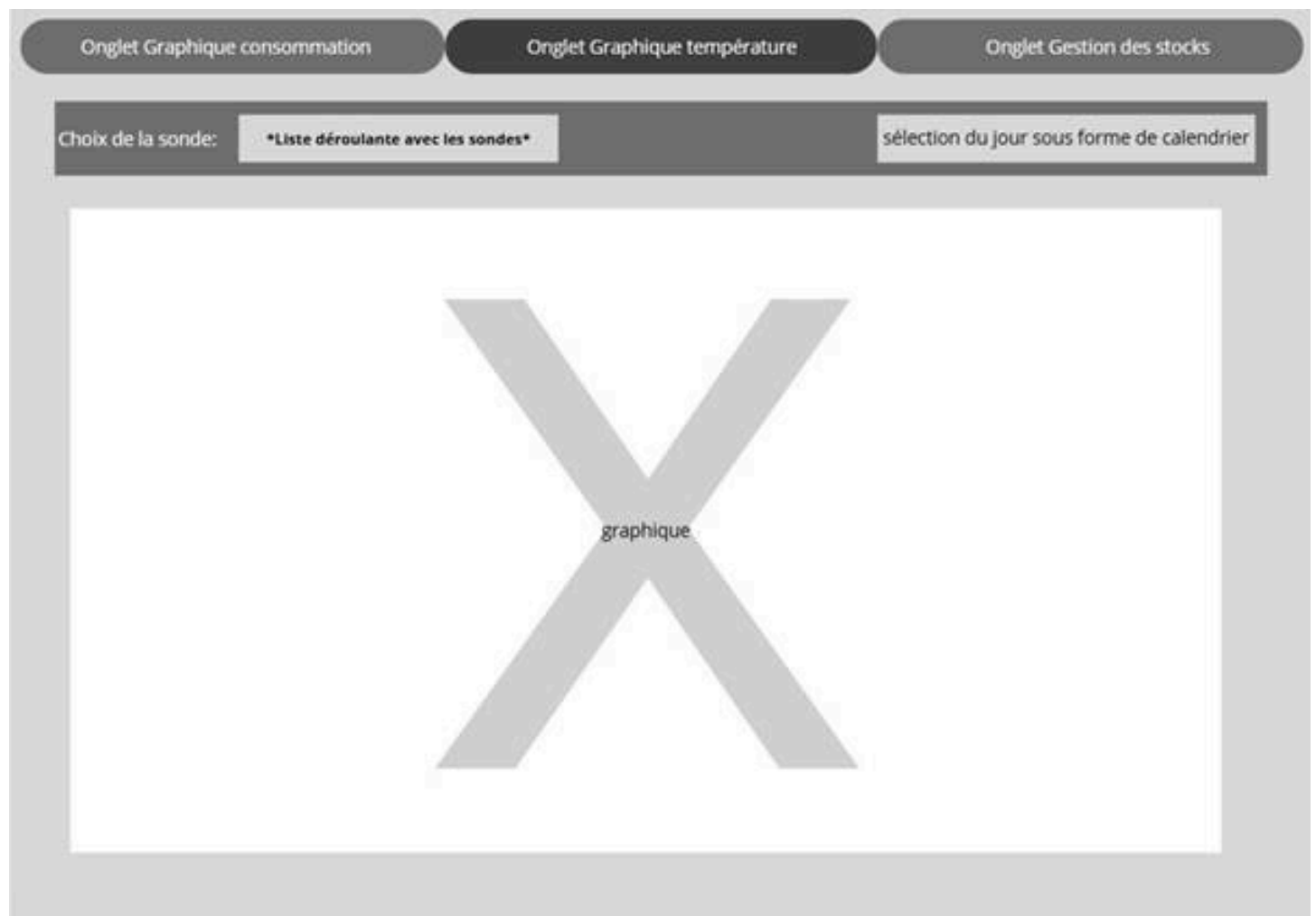


Figure 2 : Onglet température

• Parcours type pour un employé qui souhaite gérer les stocks

L'employé et le gérant du site partagent la même interface, à la seule différence que l'employé n'utilisera que le troisième onglet. L'onglet "Gestion des stocks". L'employé peut alors effectuer trois actions différentes :

- Ajouter un aliment (bouton de gauche) : En appuyant sur le bouton "Ajouter un aliment", l'interface va demander à l'utilisateur le nom de l'aliment à renseigner ainsi que la quantité et l'unité de ce dernier (ex : kg pour des patates et L pour du lait).
- Supprimer un aliment (bouton de droite) : L'employé doit d'abord sélectionner un aliment pour pouvoir le supprimer. Aucune action particulière de sa part n'est attendue.
- Modifier le stock d'un aliment (bouton du milieu) : Le dernier bouton permet, lorsque l'on a sélectionné un aliment en appuyant dessus, de modifier la quantité renseignée.

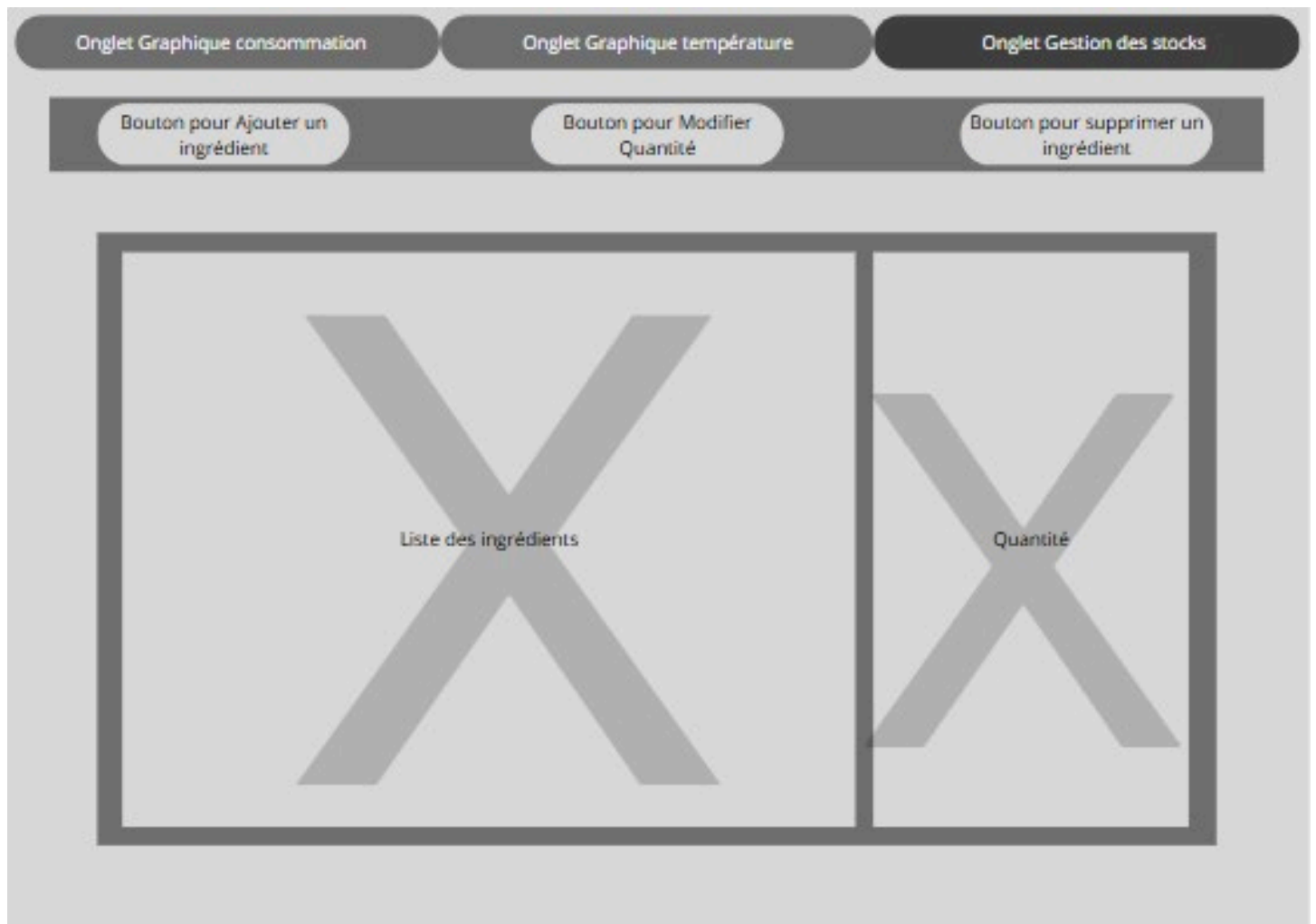


Figure 3 : Onglet gestion des stocks

2 – Maquette graphique

Comme indiqué précédemment, notre application est divisée en 3 onglets : un onglet pour le suivi de la consommation d'énergie, un pour le suivi des températures des différentes sondes et un dernier onglet dédié à la gestion des stocks. Dans cette partie, nous allons nous concentrer sur la maquette graphique, c'est-à-dire l'apparence qu'aura notre application une fois terminée.

Premier onglet :

Dans le premier onglet, celui de la visualisation de la consommation énergétique, nous avons une liste déroulante dans laquelle nous sélectionnons un appareil. En dessous, un graphique affiche les données de consommation de l'appareil sélectionné, heure par heure. Nous avons également un calendrier qui permet de choisir le jour pour lequel nous voulons afficher les données (par défaut, le jour sélectionné est la date du jour).

Suivi énergétique / stocks

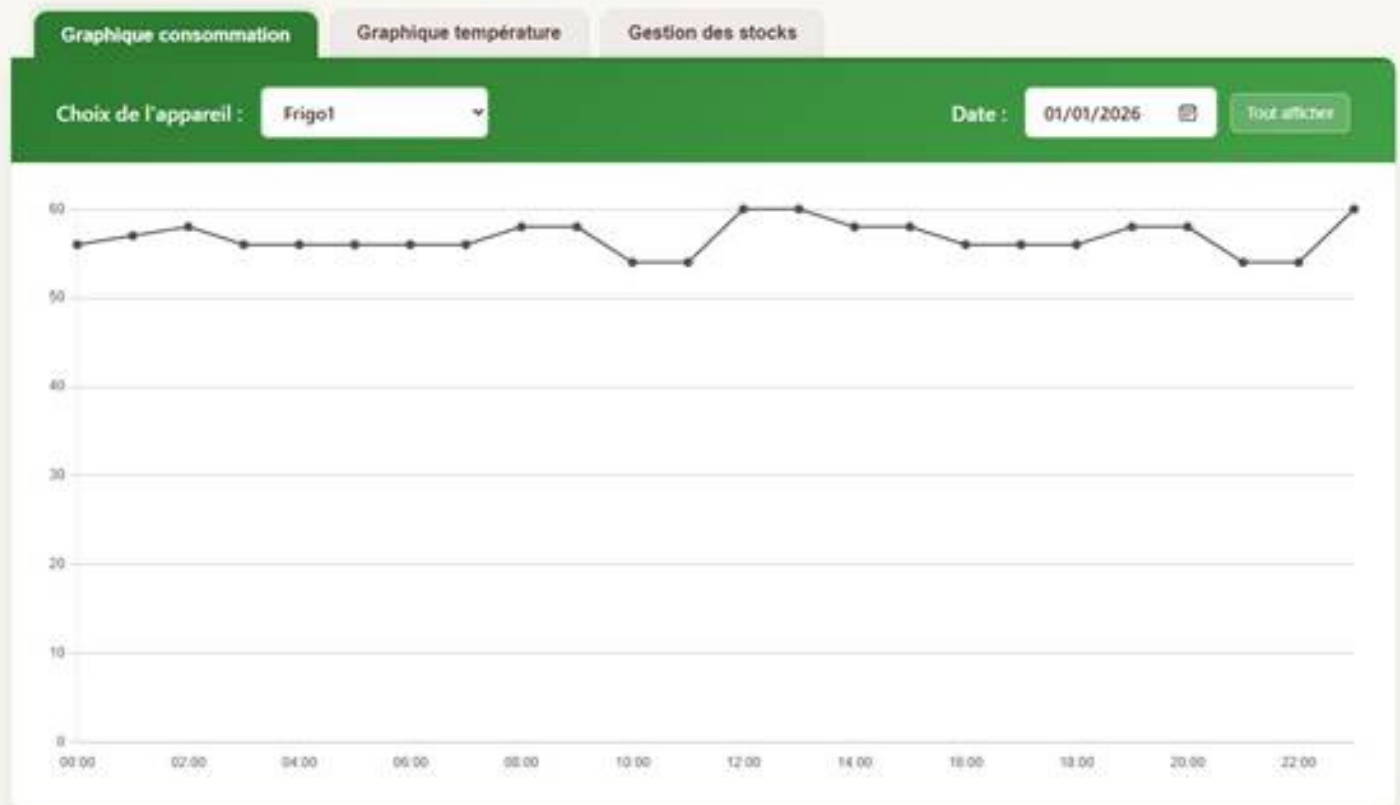


Figure 4 : Prototype / Onglet consommation

Deuxième onglet :

Le deuxième onglet reprend la même logique que le premier. Au lieu de sélectionner un appareil, nous sélectionnons une sonde. Le graphique affiché est alors celui des températures liées à cette sonde, heure par heure. Nous pouvons également sélectionner le jour dont nous souhaitons visualiser les données.

Suivi énergétique / stocks

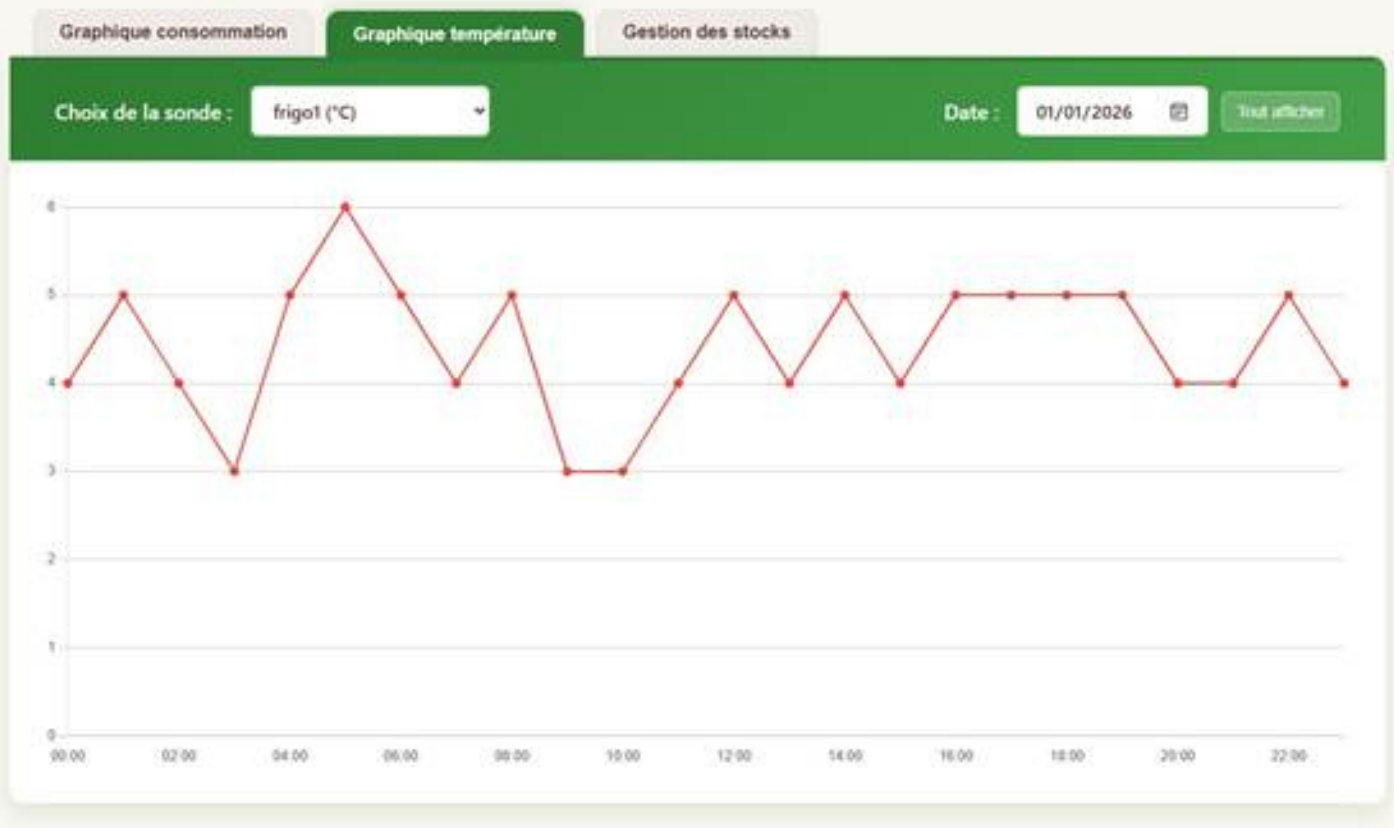


Figure 5 : Prototype / Onglet température

Troisième onglet :

Dans cet onglet, nous pouvons gérer le stock de notre restaurant et afficher les ingrédients ainsi que leurs quantités dans différents tableaux. Un bouton permet d'ajouter un ingrédient : au clic, il nous est proposé d'ajouter l'ingrédient et sa quantité. Un autre bouton permet de modifier la quantité d'un ingrédient sélectionné. Enfin, nous pouvons supprimer un ingrédient en cas de besoin.

Suivi énergétique / stocks

Graphique consommation

Graphique température

Gestion des stocks

AJOUTER UN INGRÉDIENT

MODIFIER QUANTITÉ

SUPPRIMER UN INGRÉDIENT

Liste des ingrédients

LISTE DES INGRÉDIENTS

tomate

Pomme de terre

œufs

grillade de bœuf

magret de canard

lardons

salade

miel

Quantité

QUANTITÉ

100 u

46 kg

130 u

75 u

78 u

26 kg

10 kg

5 L

Figure 6 : Prototype / Gestion des stocks